

Проблем 1. Пливање

У школи је организован турнир у пливању. Мали Коста се пријавио са намером да победи у дисциплини краул. Турнир је био организован у оквиру неколико трка и после сваке трке знамо ко је био први, други и трећи. На крају се само ови резултати узимају у обзир при одређивању шампиона, тако што прво место носи A бодова, друго место носи B , док треће место носи C бодова. Сумирају се бодови за сваког такмичара и такмичар који има највише бодова је шампион. Како бодове сабира Костин најбољи другар из клупе, он жели да одреди такав начин бодовања - да Коста буде шампион ове године.

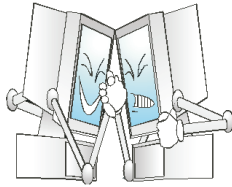
На такмичење се пријавило n ученика, при чему Коста носи капицу са редним бројем један. Турнир се одвија у оквиру m трка и за сваку од трка се знају тројица првопласираних. Максималан број бодова за прво место је p , док је минималан број бодова за треће место 1 ($p \geq A > B > c \geq 1$). Број бодова за друго место мора бити строго мањи од броја бодова за прво место и такође број бодова за треће место мора бити строго мањи од броја бодова за друго место. Уколико је могуће, одредити природне бројеве A , B и C , тако да је број бодова које је Коста освојио већи или једнак броју бодова било ког другог такмичара.

Улаз. (Улазни подаци се налазе у датотеци `plivanje.in`) У првом реду улазне датотеке се налазе три природна броја: n , m и p . Број n представља број такмичара (пливача), а m је укупан број трка. Број p је максималан број бодова које неки такмичар може да освоји у једној трци. У сваком од следећих m редова дата су три различита природна броја између 1 и n , који представљају редне бројева такмичара који су трку завршили на првом, другом и трећем месту.

Издаз. (Издазне податке уписати у датотеку `plivanje.out`) У првом и једином реду штампати бројеве A , B и C , раздвојене размаком. У случају да постоји више решења, штампати било које од њих. Уколико решење не постоји, штампати `-1`.

Ограничења.

- $3 \leq n \leq 1000$,
- $1 \leq m \leq 20000$,
- $3 \leq p \leq 50$,
- временско ограничење за извршавање програма је 1s.



Пример 1.

```
plivanje.in
3 5 10
1 3 2
3 1 2
1 2 3
3 1 2
3 1 2
```

```
plivanje.out
10 8 2
```

Објашњење. Коста је био два пута први и три пута другопласирани на турниру. Ако се за прво место добија 10 бодова, за друго 8, а за треће 2 бода, тада ће такмичари имати редом $20 + 24 + 0 = 44$, $0 + 8 + 8 = 16$ и $30 + 8 + 2 = 40$ бодова, па је шампион Коста са редним бројем један.

Пример 2.

```
plivanje.in
5 7 5
1 4 2
2 3 4
5 3 2
3 2 1
4 1 2
1 3 2
2 1 4
```

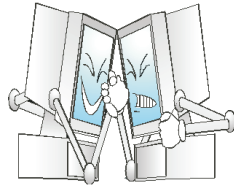
```
plivanje.out
5 4 1
```

Пример 3.

```
plivanje.in
8 4 4
3 8 1
2 7 3
2 1 6
5 4 1
```

```
plivanje.out
-1
```

Објашњење. Ако се за прво место добија 3 бода, за друго место 2 и за треће 1 бод - онда ће такмичар број 2 имати 6 бодова, а Коста 4 бода. Ако се за прво место добија 4 бода, тада ће такмичар са редним бројем 2 имати 8 бодова, а Коста највише $3 + 4 = 7$ бодова. Ни у једном случају Коста не може да победи.



Проблем 2. Викендица

Драганчету је досадило да живи у граду па је решио да сагради себи викендицу, у којој ће у миру и тишини моћи да спроводи своје програмерске идеје. Како је Драганче веома вредан, решио је да сам направи нацрте његове куће из снова. Но, на почетку је требало да одреди место на коме ће се градити. Како Драганче није баш богат (математика и програмирање није било исплативо како је он мислио), одредио је максимални буџет који може да потроши за куповину плацева. Наравно, жели да викендица буде што лепша, па је одредио и доњу границу буџета. Сад је у недоумици где да је сагради. Помозите Драганчету да одреди број могућих места за градњу викендице.

Места на којима Драганче може да гради викендицу дата су у облику правоугане матрице, где поља представљају плацеве. За сваки плац је дата његова цена. Викендица је у облику правоугаоника. На правоугаонику се може саградити викендица уколико сума њених парцела упада у границе буџета.

Улаз. (Улазни подаци се налазе у датотеци `vikendica.in`) У првом реду се налазе четири броја: n , m , A и B , који представљају број врста и колона матрице, доњу границу и горњу границу буџета. У наредних n редова налази се по m ненегативних целих бројева који представљају цене плацева.

Издаз. (Издазне податке уписати у датотеку `vikendica.out`) У првом и једином реду штампати број могућих позиција викендице.

Ограничења.

- $1 \leq n, m \leq 150$,
- $0 \leq A \leq B \leq 10^9$,
- цене плацева су у опсегу $[0, 10^4]$,
- број решења не прелази 2^{30} ,
- укупна сума цена свих плацева је мања од 2^{30} ,
- временско ограничење за извршавање програма је 1s.

Пример 1.

```
vikendica.in                                vikendica.out
3 3 2 3                                      7
1 0 0
0 1 0
0 0 1
```

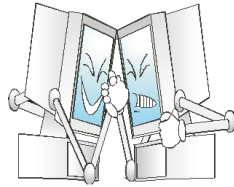
Објашњење. Могуће позиције за викендицу (представљене са 'o') су следеће:

```
o o x o o o o x x x x o o x x x o o o
o o x o o o o x o o x o o o o o o o o
x x x x x x o o x o o x o o o o o o o
```

Сума елемената у означеним правоугаоникима је у опсегу $[2,3]$.

Пример 2.

```
vikendica.in                                vikendica.out
3 4 7 10                                      16
1 3 4 2
3 4 5 2
1 3 4 1
```



Проблем 3. Кредит

Професор Ђурић: Халоооо!

Драганче: Добар дан, професоре, Драганче је овде.

Професор Ђурић: Шта рааадиш, момчино?

Драганче: Управо шетам парком, и кад сам стигао до фонтане, добио сам идеју за онај задатак о коме смо јуче дискутовали...

Професор Ђурић: А јеее ли.

Тако се наставио телефонски разговор, који је трајао K минута, и ко зна колико би још трајао да Драганчету није нестало кредита. Драганче је одмах кренуо да купи додатни кредит, пошто зна локације свих трафика у парку, и кренуо је у ону до које му треба најмање времена. Чим стигне и уплати кредит, позваће професора поново. Професор Ђурић жели да квалитетно организује своје време и потребна му је процена колико Драганчету треба времена да поново позове.

Улаз. (Улазни подаци се налазе у датотеци `credit.in`) И Ђурић и Драганче тачно знају мапу парка, која је дата у облику правоугаоне матрице димензија $N \times M$, где су N и M бројеви задати у првом реду датотеке. У другом реду је број K . У сваком од наредних N редова се налази по M знакова, који означавају поља на мапи. Дозвољени знакови су 'F', 'X', '.' и 'T'. Знак 'F' означава фонтану. Све препреке (нпр. језеро, стадион, цвеће...) означене су са 'X', док су са '.' означени делови парка слободни за шетњу. Са 'T' су такође означени делови парка где се може проћи, и у њима се налази трафика где се може уплатити кредит. Драганче се на почетку разговора налази код фонтане. У једном минуту он може прећи у неко од суседних поља (исток, запад, север или југ), а може остати и ту где јесте. У току разговора он се шета насумично. Када буде кренуо до трафике, неће правити паузе, већ ићи најкраћим путем док не стигне до ње.

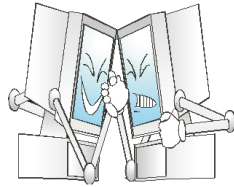
Излаз. (Излазне податке уписати у датотеку `credit.out`) Професор Ђурић не зна где се Драганче шетао у току разговора и жели да процени минимално и максимално време које је потребно Драганчету да дође до трафике са кредитом. У једином реду излазне датотеке треба исписати два броја, MIN и MAX , одвојена размаком; они означавају минимални и максимални број минута потребан да Драганче стигне до неке од трафика.

Ограничења.

- бројеви N и M нису већи од 200,
- K није веће од 500,
- број трафика није већи од 1000,
- знак 'F' се налази тачно једном у улазној датотеци,
- временско ограничење за извршавање програма је 1 s.

Парк је такав да Драганче увек може стићи до неке од трафика. У делу са фонтаном се не налази трафика.

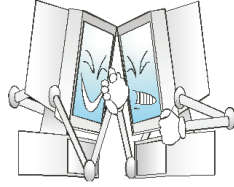
Напомена. Ако је барем један од бројева MIN или MAX тачан, добићете 50% поена за тај пример.



Пример 1.

```
kredit.in          kredit.out
5 5                2 5
3
X..XF
.X...
T..X.
X.T..
...T.
```

Објашњење. После 3 минута разговора, Драганче ће моћи да се приближи некој од трафика и за само 2 минута стигне до ње, а могуће је и да остане све време крај фонтане, одакле ће му требати 5 минута до најближе трафике.



Проблем 4. "LAN party"

Сви љубитељи компјутерских игрица знају какво је уживање победити противника са што већом разликом. У том погледу мали Коста је велики хедониста.

Он и његови другари воле да се повремено окупљају и играју игрице по цео дан. Они се поделе у два тима и играју n партија, при чему се ниједна партија не завршава нерешено. Како би његов тим што боље прошао у укупном исходу, Коста вешто смишља разне начине рачунања коначног резултата. Омиљена стратегија му је следећа: он n партија које су одиграли подели у највише k интервала, тако да сваки интервал чине узастопне партије и свака партија припада тачно једном интервалу. За сваки интервал потом рачуна победника тако што гледа чији тим је победио више пута у партијама тог интервала (ако су тимови имали исто победа онда интервал нема победника). Коначан резултат Коста онда исказује победама у интервалима, а не у партијама.

Ваш задатак је да помогнете Кости да направи такву поделу партија на интервале да његов тим направи што већу разлику (по његовом рачунању), при чему му је противнички тим задао ограничење да не буде више од k интервала.

Ако му помогнете, Коста ће вас наградити бесплатном улазницом за све утакмице америчког фудбала које се одржавају на помоћном терену иза стадиона.

Улаз. (Улазни подаци се налазе у датотеци `lanparty.in`) У првом реду улазне датотеке налазе се природни бројеви n и k , раздвојени размаком. У наредном реду се налази низ од n слова при чему је свако од слова 'W' или 'L'. Између тих слова нема размака. Слово на i -тој позицији означава исход i -те партије: 'W' ако је победио Костин тим, а 'L' ако је победио противнички тим.

Излаз. (Излазне податке уписати у датотеку `lanparty.out`) На излаз запишите максималну разлику коју Костин тим може остварити поделом партија на интервале.

Ограничења.

- $1 \leq n \leq 100$,
- $1 \leq k \leq n$,
- временско ограничење за извршавање програма је 1 s.

Пример 1.

<code>kosta.in</code>	<code>kosta.out</code>
11 5	2
LLWWLWLWLWW	

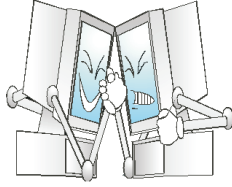
Објашњење. Један од могућих начина да се оствари разлика 2 је да се формира следећих 5 интервала:

- интервал 1 чине партије 1 и 2 - победник у овом интервалу је противнички тим,
- интервал 2 чине партије 3, 4 и 5 - победник у овом интервалу је Костин тим,
- интервал 3 чини само партија 6 - победник је опет Костин тим,
- интервал 4 чине партије 7 и 8 - оба тима имају једнако победа у партијама, па овај интервал нема победника,
- интервал 5 чине партије 9, 10 и 11 - победник и у овом интервалу је Костин тим,

Пример 2.

<code>kosta.in</code>	<code>kosta.out</code>
13 5	-1
LLLWLLLWLLWLL	

Објашњење. Водите рачуна да највећа разлика може да буде и негативна.



Проблем 5. Подниз

Када сте успешно помогли Драганчету да брзо реши задатак Гаус, професор Ђурић се стварно наљутио и решио да за следећи час смисли стварно тежак задатак. Професор Ђурић је сав срећан дошао сутрадан у школу, и дао ђацима следећи задатак. На табли је написао низ бројева, и тражи да пронађу најкраћи низ чији су елементи бројева из скупа $\{1, 2, \dots, n - 1, n\}$ (при чему се у низу могу понављати исти бројеви) који није подниз низа написаног на табли. Сва деца су се умирила и професор Ђурић је седео задовољно, очекујући да је миран бар за следећих неколико година. Али превидео је две мале ствари: да је мали Драганче паметнији од остале деце, а и да се дружи са вама, добрим програмерима. Драганче мисли да је смислио начин да нађе дужину траженог низа, али не и који је то низ, па вас је зато позвао у помоћ. Прво треба да нађете дужину тог низа, да би Драганче проверио своје решење, а затим не би било лоше ни да нађете тај низ. Да ли ће професор Ђурић поново бити љут, остаје на вама да се види.

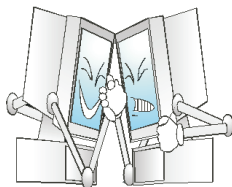
Улаз. (Улазни подаци се налазе у датотеци `podniz.in`) У првом реду налазе се два броја, n и m , где је m број елемената низа записаног на табли. Бројеви написани на табли и бројеви у траженом (под)низу су из скупа $\{1, 2, \dots, n - 1, n\}$. У следећих m редова налази се по један број и они представљају бројеве записане на табли.

Излаз. (Излазне податке уписати у датотеку `podniz.out`) У првом реду написати k - број елемената траженог низа. У следећих k редова написати редом по један елемент траженог низа. Ако има више низова исте дужине, написати онај који је први лексикографски. Низ b је подниз низа a ако постоји низ t тако да за свако i важи $b(i) = a(t(i))$, $t(i) < t(i + 1)$ (дакле, низ b се може добити од низа a избацивањем неких елемената). Тј. од низа 1213 поднизови су нпр. 12, 23, 123, а нису 31, 22. За тачно израчунат број k добија се 40% поена на том тест примеру.

Ограничења.

- $n, m \leq 1000000$,
- временско ограничење за извршавање програма је 1s.

Напомена. У језицима C/C++ користите библиотеку `stdio`, а не `iostream`, јер `iostream`-у треба више од 1 секунде да учита 1000000 бројева.



Пример 1.

podniz.in	podniz.out
2 3	2
1	2
2	2
1	

Пример 2.

podniz.in	podniz.out
2 4	3
1	1
2	1
2	1
1	

Пример 3.

podniz.in	podniz.out
3 10	4
1	1
2	1
3	2
3	2
2	
1	
1	
2	
3	
1	